System Commands, Editors and Utilities

The following topics are covered:

- Enhanced System Commands
- Removed System Commands
- New Utilities
- Enhanced Editors and Utilities
- Utility Activation
- Application Programming Interfaces USR* in Library SYSEXT
- New Application Programming Interfaces in Library SYSEXT
- Software AG Editor
- Debugger Enhancements

Enhanced System Commands

The following Natural system commands have been enhanced for Natural Version 4.1:

- CATALL
- COMPOPT
- DUMP
- LAST
- LIST
- STRUCT

CATALL Command

The CATALL command provides the following enhancements:

- The error list provided by the CATALL command has been expanded to include additional information on the causes of CATALL processing errors.
- A user exit has been provided which allows you to change the default settings of the main CATALL screen.

COMPOPT Command

The COMPOPT command provides the following enhancements:

- With the MASK option in a logical condition, a valid year value must be in the range 0000 - 2699; with the MOVE EDITED statement, a valid year value must be in the range 1582 - 2699. Consequently, a year value found valid by a MASK option might lead to an error in a subsequent MOVE EDITED statement. To avoid this situation, the COMPOPT command provides a new option, MASKCME, which allows you to determine whether the year range to be considered valid by the MASK option is to begin with 0000 or 1582.
- With Natural Version 2.3, the internal handling of assignments between two Format N variables of the **same** length was changed, so as to be consistent with the internal handling of assignments between Format N variables of different lengths. One side effect of that change was that when these variables were redefined, this could in some cases lead to different results.
  With Version 4.1, the COMPOPT command provides a new option, NMOVE22, which allows you to determine whether the internal handling of assignments between Format N variables of the same length is to be inconsistent as in Version 2.2 or consistent as in all subsequent versions (the latter will be the default).
  The NMOVE22 option replaces the corresponding special-purpose Zaps that were supplied for Natural Versions 2.3 and 3.1. To provide upward compatibility at runtime for objects cataloged with previous Natural Versions **without** the Natural Optimizer Compiler, a Zap will be provided with Natural Version 4.1. This Zap will not be provided for subsequent Natural versions. Objects cataloged with previous Natural Versions with the Natural Optimizer Compiler are not affected and will continue to work as before. Affected objects should be recataloged as soon as possible with Natural Version 4.1, with NMOVE22=ON being set.
- The option V31COMP allows to reject syntax constructions that are supported by Version 4.1, but not by Version 3.1. This option will be available only with Natural Version 4.1 to allow a smooth transition. It will be removed again with the next major release of Natural after Version 4.1.
- The option TQMARK to check for the translation of quotation marks.
- With the option PCHECK set to ON, the compiler checks the number, format, length and array index bounds of the parameters that are specified in a CATALL statement. Also, the OPTIONAL feature of the DEFINE DATA PARAMETER statement is considered in the parameter check.

Note:
See also Removed System Commands, for information on the V22COMP option of the system command COMPOPT.

DUMP Command

The DUMP command provides the following enhancements:

- An SRCE function to show the inventory of source changes applied per product (similar to the ZAPS function).
- An SVAR function to show TP monitor or operating system dependent system variables and additional information.

LAST Command

The LAST command provides the following enhancement:

In previous Natural versions, "LAST *" displayed a window showing the last 9 system commands that were issued. With Natural 4.1, this window shows the last 20 system commands.

LIST Command

The LIST command provides the following enhancement:

- When the LIST command displays a list of objects, these objects are in alphabetical order of their names. With Natural 4.1, it is also possible to sort the list of objects by type, date, etc.

STRUCT Command

The STRUCT command now supports two-digit field levels.

Removed System Commands

The following Natural system command or command option was dropped as of Natural Version 4.1:

COMPOPT Command

The V22COMP option of the system command COMPOPT (Allow old Version 2.2 Syntax) has been removed. Consequently, this option will also be invalid for the NTCMPO macro, the CMPO profile parameter and the OPTIONS statement.

SYSBUS

The Natural system command SYSBUS is no longer available. Instead, you use the system command BUS which performs the same function.

New Utilities

The following new Natural utility is provided with Natural Version 4.1:

Natural Object Handler

The Natural Object Handler processes objects for the purpose of application distribution. This utility is invoked with the system command SYSOBJH and combines the functionality currently provided by the utilities SYSTRANS and NATUNLD/NATLOAD. The utilities SYSTRANS and NATUNLD/NATLOAD will cease to be available with the next major release of Natural (following Version 4.1).

The Natural Object Handler enables you to

- unload objects in the source environment to work files, and then load these objects from work files into the target environment.
- process Natural programming objects, resources, DDMs, error messages, Natural-related objects, Natural command processors, external objects, and Adabas FDTs.
- perform unloading and loading in an internal format (as with the NATUNLD/NATLOAD utility) or in a transfer format (as with the SYSTRANS utility).
- process work files that were created with the utilities SYSTRANS and NATUNLD/NATLOAD.

Work files created with the Natural Object Handler in transfer format can be processed by the utility SYSTRANS on all platforms.

Enhanced Editors and Utilities

The following enhancements are provided with Natural 4.1:

- Data Area Editor
- Map Editor
- NATLOAD/NATUNLD
- SYSBPM
- SYSDDM
- SYSMAIN
- SYSNCP
- SYSPARM
- SYSRDC
- SYSTP
- SYSWEB

Data Area Editor

The Natural data area editor provides full support of the following enhancements of the Natural language:

- Alphanumeric and binary variables with a length of up to 1 GB
  (See also Miscellaneous Changes and Enhancements, Size of Alphanumeric and Binary Variables.)
- Dynamic variables
  (See also Programming Language Enhancements, Dynamic Variables.)
- Array indices of up to 10 digits; the maximum index value is 1073741824, the maximum size of an array may now be 1 GB
  (See also Miscellaneous Changes and Enhancements, Size of Data Elements.)
- Optional parameters
  (See also Programming Language Enhancements, Optional Parameters.)

The following additional enhancements are provided:

- The number of possible field levels in a data area has been increased from 9 to 99.
- It is now possible to define object handles also within a global data area. With Natural Version 3.1, object handles could only be defined within a local or parameter data area.
- It is now possible to define initial values for object handles.

The user interface has been adapted to support the features described above.
The .E line command (used to define initial values and edit masks) has been enhanced. On the separate screen that is displayed after the .E line command has been entered, the new command code A has been added to allow for the definition of long indices.

New features not available with Natural Version 3.1 require that the data area is stored in the FUSER system file using a new and extended source format. As long as no new features are used, data areas are stored using the old Natural Version 3.1 compatible source format by default to allow for sharing the data area between a Natural Version 3.1 and Natural Version 4.1 environment.

The V31COMP compiler option may be used to ascertain that a data area that is edited and cataloged with the Natural Version 4.1 data area editor can still be cataloged with Natural Version 3.1.

Map Editor

The following Natural Version 4.1 features have been added to the Natural map editor:

- **Support of Long and Dynamic Variables**

  Long and dynamic alpha variables can be used in the map editor. This includes the feature that the length of these variables will be automatically adjusted by the map editor to the length fitting on the map by the use of the AL parameter.

- **Support of Large Arrays**

  The array dimensions have been adjusted to the new limits available in the Natural 4.1 system.

- **Support of Selection Boxes**

  The new feature "Selection Boxes on Mainframe" has been added to the map editor. In the Extended Field Editing Menu, it is only required to specify the name of the selection-box array, the rest is done automatically by the map editor.

NATLOAD/NATUNLD Utility

The NATLOAD/NATUNLD utility provides the following enhancement:

- If objects are to be unloaded/loaded via Entire Connection and Entire Connection has not been activated, the terminal command %+ is automatically issued to activate Entire Connection.
- A new user exit LOADEX03 (source code L-S-EX03) has been provided which can be invoked as soon as NATLOAD processing has finished.
- A new option FIXEDLENGTH in the unload-with-clause causes all records to be written to the work file with a fixed length of 252 bytes.
- A new option VERSIONCHECK in the load-object-with-clause can be used to make sure that the version of the cataloged objects to be loaded is compared with the current Natural version.
- New online commands for report processing have been introduced (see Condition Codes and User Exits in Batch Mode in the NATLOAD/NATUNLD documentation).

SYSBPM Utility

The SYSBPM utility provides the following changes and enhancements:

- The utility main menu and the LIST function were subject to major changes.
- New utility direct commands WRITE, SORT, DISPLAY CDIRECTORY have been added.
- The direct command SELECT BUFFERPOOL is now available as a function that can be selected from the SYSBPM menu. This also eliminates the need for using the DISPLAY BUFFERPOOL and RESET BUFFERPOOL direct commands (the display function is implicite and a reset can be accomplished by selecting a different buffer pool from the selection list).
- The following utility direct commands have been changed:
  - DISPLAY CLIST replaces DISPLAY CINDIVIDUAL.
  - DISPLAY HEX replaces DISPLAY OBJECT.
  - DISPLAY LIST replaces DISPLAY INDIVIDUAL.
- A new SYSBPM user interface provides the option to sort the objects in the buffer pool by various criteria (for example, use count) in batch mode.

SYSDDM Utility

The SYSDDM utility provides the following enhancement:

- Support of large und dynamic variables for DB2 DDMs.
- Support of access to DB2 tables on UNIX platforms.
  For details, see SQL Services in DDM Generation in the Natural for DB2 documentation.

SYSMAIN Utility

The SYSMAIN utility provides the following enhancement:

- While the function Find Programming Object is being executed, SYSMAIN displays a window indicating the name of the object being searched for. With Natural Version 4.1, this window will also indicate whether the source form or cataloged form of an object is being searched for.
- A new online command FINDFIRST has been introduced that causes the FIND function to stop when the first object is found. See Commands Issued to SYSMAIN in the SYSMAIN documentation for more information.
- In user exit MAINEX11 (source SM-UX-11), a new flag has been introduced that enables suppression of the FIND Objects window in the FIND function.

SYSNCP Utility

The user exits of the SYSNCP utility have been modified. Please use these new user exits and adapt them to your requirements. See also Utility Activation.

SYSPARM Utility

The SYSPARM utility provides the following enhancement:

- You can execute SYSPARM utility functions in direct-command mode or batch mode: see Direct Commands and Batch Processing in the Natural SYSPARM Utility documentation.
- With Natural 3.1, you could only maintain parameter profiles stored on and applying to the current FNAT system file. With Natural 4.1, it is also possible to maintain parameter profiles of other FNAT system files.
- You can enter a commentary text in each editor line, or spread a comment over as many lines as you like.

SYSRDC Utility

The SYSRDC utility provides the following enhancements:

- New function Trace Record by Number (function code N)
- New function to show the status of trace recording (function code C)

SYSTP Utility

The SYSTP utility provides the following enhancement:

- A new SYSTP function allows you to cancel Natural user sessions depending on their last-activity date.

SYSWEB Utility

A new version of SYSWEB based on Natural for Windows/UNIX Version 6.1 has been provided.

Utility Activation

From Natural Version 4.1, Natural invokes a Natural utility without performing a logon to the corresponding utility library in the FNAT system file. This applies to all utilities listed in Support of Activation without Logon below. As a result, Natural preserves the global data area (GDA) and/or application-independent variables (AIV). For resetting the GDA and/or the AIVs, see the parameter FREEGDA below. The current user library and the steplib settings are maintained.

To preserve the settings of your application environment, do **not** log on to a utility library. Instead, invoke a utility by using the Natural system command that corresponds to the utility.

After terminating a utility, you will be returned to the library from which you invoked the utility. However, if you explicitly log on to a utility library before invoking the utility, you will stay in this (utility) library after utility termination.

The behavior described above can have an impact on Natural Version 4.1 batch procedures that contain logon instructions to utility libraries.

For information on how to control access to a utility in a Natural Security environment, see Protecting Utilities in the Natural Security documentation.

Below is information on:

- Support of Activation without Logon
- Source Editing
- FREEGDA Parameter
- SYSNCP User Exit

Support of Activation without Logon

Below are the Natural utilities that support utility activation (without logon to the corresponding utility library) by using the corresponding Natural system commands:

- NATLOAD
- NATUNLD
- SYSADA
- SYSBPM
- SYSDB2
- SYSDDM
- SYSEDT
- SYSERR
- SYSMAIN
- SYSNCP
- SYSAFOS
- SYSSAFOS
- SYSOBJH
- SYSPARM
- SYSPOOL
- SYSRPC
- SYSSEC
- SYSSQL
- SYSTP (including SYSMON, SYSSWAP, SYSFILE, BUS)
- SYSTRANS
- SYSUNLD

Below are the Natural utilities/Natural system commands that do **not** support utility activation without logon to the utility library. Invoking these utilities will still cause an implicit logon to the utility library and user settings can be lost.

- SYSEXT
- SYSEXV
- SYSRNM (to apply the utility activation without utility library logon requires a new Review Natural Monitor version; see Natural and Other Software AG Products in the section General Information of these Release Notes)
- All other Natural add-on and Software AG products using the Natural FNAT system file.

Source Editing

Source editing in utility libraries still requires a logon to the corresponding utility library.

FREEGDA Parameter

The profile parameter FREEGDA controls whether current user global data area (GDA) and application-independent variables (AIV) are to be reset when a utility is invoked by using the Natural system command that corresponds to the utility's name.

Note that the preservation of the current user GDA and AIV variables will increase the data size accordingly and can lead to thread size problems under certain operating systems. In this case, you can use the system command LOGON *library* to force a logon to the utility library.

See also FREEGDA in the Natural Parameter reference documentation.

SYSNCP User Exit

With utility activation under Natural Version 4.1, User Exit NCP-USR1 has changed. If you apply this user exit, use the new NCP-USR1 program and adapt it to your requirements.

Application Programming Interfaces USR* in Library SYSEXT

With Natural Version 4.1, the USR* programs (APIs, previously named "user exits", see also note on terminology change in the section Documentation) from the library SYSEXT delivered will run in a special mode. As a result, the USR* programs need no longer set further steplibs to execute related Natural objects for processing. This will reduce the impact on the Natural buffer pool search logic and will improve performance significantly if application programming interfaces are used extensively within user written applications.

To introduce this mode, the application programming interfaces must be cataloged with Natural Version 4.1. This implies that the application programming interfaces cannot be executed with Natural Version 3.1. This behavior is different to the previous Natural version and will have some impact on the migration path of applications that call application programming interfaces.

Use of USR* Programs

Usually the access of USR* programs by an application requires that the application programming interfaces are copied from the library SYSEXT to the application libraries on the FUSER system file or to the library SYSTEM on the FUSER system file or the library SYSTEM on the FNAT system file or any other library which is defined as steplib for the application. Library SYSEXT can also be used as steplib. Because, as of Natural Version 4.1, the application programming interfaces delivered will always be cataloged with the latest Natural version, we recommend that the application programming interfaces should reside on the FNAT system file. This will ensure that the right version is executed and separates user-written applications from modules provided by Software AG.

If applications that call application programming interfaces should run with both Natural Version 3.1 and Natural Version 4.1, the application programming interfaces delivered with the corresponding Natural version must be used.

The following scenarios may be considered:

- Using a Version 3.1 FUSER File for Natural Version 3.1 and 4.1
- Using a Version 3.1 FUSER File for Natural Version 4.1 only
- Using a New FUSER File for Natural Version 4.1
- Migration

Using a Version 3.1 FUSER File for Natural Version 3.1 and 4.1

If the same FUSER system file is to be used in a Natural Version 3.1 and Version 4.1 environment in parallel, the following steps are recommended:

- Remove all USR* modules you have copied from the library SYSEXT into application libraries on your FUSER system file.
- In both environments, copy the used USR* modules from the library SYSEXT to the library SYSTEM on the corresponding FNAT system file.

Alternatively, the USR* modules can be moved to another system library on FNAT, which then must be defined as steplib, or the library SYSEXT can be used as steplib for the applications. Then, automatically in both environments, the right versions of the application programming interfaces are executed.

Using a Version 3.1 FUSER File for Natural Version 4.1 only

If you want to use the existing Natural Version 3.1 FUSER system file and you do not want to share the FUSER system file, then it is still possible to replace all USR* modules you have copied from the library SYSEXT into application libraries with the new USR* objects from the Version 4.1 library SYSEXT.

However, the preferred way is to remove all application programming interfaces on the FUSER system file and copy the used application programming interfaces from the library SYSEXT to the library SYSTEM of the FNAT system file or use an SYS library on FNAT as steplib.

Using a New FUSER File for Natural Version 4.1

If you want to port existing applications to a new FUSER file, copy all applications objects but no USR* objects originating from Software AG to the new FUSER system file. Then proceed as described in the scenario above.

Migration

The function FIND of the SYSMAIN utility can be used to search for all USR* modules stored in a specific library on the FUSER system file or across the whole system file. In addition, Predict cross-reference data can be used to determine all referenced application programming interfaces.

New Application Programming Interfaces in Library SYSEXT

The library SYSEXT provides the following new application programming interfaces (APIs, previously named "user exits", see also note on terminology change in the section Documentation):

API	Description
USR2035	Set SSL parameter string for communication via RPC.
USR2073	Ping or terminate an RPC server.
USR4001	Set Natural profile parameter PROGRAM dynamically.
USR4002	Retrieve variables of the current system.
USR4003	Retrieve Natural stack information.
USR4004	Retrieve Natural dynamic parameter.
USR4005	Read all current key settings.
USR4007	Get/set current value of profile parameter SYNERR.
USR4340	Natural Buffer Pool Interface 4340.
USR6002	Get the current values of some internal counters.

Software AG Editor

- The Software AG Editor parameter macro NTEDBP is now an integral component of the parameter module NATPARM.
- The Software AG language support has been changed. The language tables (English, Hebrew and Japanese) were moved from the macro NATEDTAB into separate macros NTEDTAB*n* (*n*=1,2,3) which are called by the source module NATCONFG.
- The Software AG Editor supports the Natural system variable *LANGUAGE for Hebrew (ULANG=33) and Japanese (ULANG=59).
- The Software AG Editor supports the changes of format for NODE and JOB-NUMBER introduced with Entire System Server Version 3.2.1.

Debugger Enhancements

- Enhanced Display Functions
- Break Processing
- Commands

Enhanced Display Functions

- The List Source function in addition supports the display on Model-4 screens.
- In the List Source function, you may set a watchpoint by setting the cursor on a variable and pressing PF18.
- The Display Variables screen was enhanced with a zoom function (PF5) to display large variable names, possibly with long indices.
- The screen for displaying the content of a single variable was enhanced to display long variables in chunks of 256 bytes after a selected position. Browsing in the content of a long variable is possible with PF keys PF22/PF23.

Break Processing

- Watchpoints can be set for long or dynamic variables. If the length exceeds 253 bytes, the comparison value of a watchpoint is restricted to the first 253 bytes.
- New watchpoint type: persistent watchpoint.
  A persistent watchpoint allows extending the scope of investigation of a variable value: whereas a standard watchpoint checks for a change of content of a variable in the scope of an object, a persistent watchpoint also checks the change of content of a variable on a subordinate level.
- New watchpoint operator: INV.
  The watchpoint operator INV allows you to check any invalid content of variables of the type N, P, D, T.

Commands

- The command DBLOG is now also available during debugging.

